

CLAIMSWHAT IS CLAIMED:

1. An apparatus, comprising:

5 a scoreboard comprising a plurality of locations adapted to store transaction identifiers, wherein a transaction comprises a first client sending a request to a second client; and

a device adapted to manage the plurality of transaction identifiers in the scoreboard.

10 2. The apparatus of claim 1, wherein the device comprises a timer adapted to approximately synchronously compare the length of time the transaction identifiers remain in the scoreboard to a predetermined latency period.

15 3. The apparatus of claim 2, wherein the timer is a free-running timer.

4. The apparatus of claim 3, wherein the free-running timer is a cyclical free-running timer adapted to return to a zero-point after the predetermined latency period.

20 5. The apparatus of claim 4, wherein the predetermined latency period ranges approximately from 6 nanoseconds to 28 seconds.

6. The apparatus of claim 2, wherein the device further comprises a fill-code generator adapted to initiate a time-out sequence.

7. The apparatus of claim 6, wherein the fill-code generator is adapted to initiate the time-out sequence when the timer notifies the fill-code generator that at least one of the transaction identifiers has remained in the scoreboard longer than the predetermined latency period.

8. The apparatus of claim 7, wherein the fill-code generator is adapted to create a fill code and transmit the fill code to the first client when notified that at least one of the transaction identifiers has remained in the scoreboard longer than the predetermined latency period.

9. The apparatus of claim 8, wherein the fill-code generator is further adapted to notify the client that a time-out occurred.

10. The apparatus of claim 1, wherein the transaction identifiers comprise a client ID, a client tag, and a first and a second timer flag.

11. The apparatus of claim 1, wherein the first client is at least one of a processor, a memory, and an I/O device.

12. The apparatus of claim 1, wherein the second client is at least one of a processor, a memory, and an I/O device.

13. An apparatus, comprising:

a scoreboard comprising a plurality of locations adapted to store transaction identifiers,
wherein a transaction comprises a first client sending a request to a second client;

5 a timer adapted to compare the length of time the transaction identifiers remain in the
scoreboard to a predetermined latency period; and

a fill-code generator adapted to initiate a time-out sequence when notified that at least
one transaction identifier has remained in the scoreboard for substantially longer
than the predetermined latency period.

10 14. The apparatus of claim 13, wherein the timer is a free-running timer.

15 15. The apparatus of claim 14, wherein the free-running timer is a cyclical timer
adapted to return to a zero-point after the predetermined latency period.

16. The apparatus of claim 15, wherein the predetermined latency period ranges
approximately from 6 nanoseconds to 28 seconds.

20 17. The apparatus of claim 13, wherein the fill-code generator is adapted to create a
fill code and transmit the fill code to the first client when notified by the free running timer that
at least one of the transaction identifiers has remained in the scoreboard longer than the
predetermined latency period.

18. The apparatus of claim 17, wherein the fill-code generator is further adapted to notify the client that a time-out occurred.

19. A method, comprising:

5 storing at least one transaction identifier in at least one of a plurality of locations in a scoreboard, wherein each transaction comprises a first client sending a request to a second client in a system;

timing a selected duration; and

10 initiating a time-out sequence if the selected duration is substantially longer than a predetermined latency period.

20. The method of claim 19, wherein timing the selected duration comprises comparing a period of a free running timer to approximately the length of time a transaction identifier remains in the scoreboard.

15 21. The method of claim 20, wherein comparing comprises detecting transaction identifiers that have been stored in the scoreboard substantially during the previous period of the free-running timer.

20 22. The method of claim 21, wherein detecting transaction identifiers that have been stored in the scoreboard substantially during a previous period of the free-running timer comprises examining a first timer flag when the free-running timer reaches a zero point.

23. The method of claim 22, wherein detecting transaction identifiers that have been stored in the scoreboard substantially during a previous period of the free-running timer further comprises setting the first timer flag when the first timer flag has not been set and setting a second timer flag when the first timer flag has been set.

24. The method of claim 23, wherein comparing further comprises determining when a transaction identifier has been stored in the scoreboard for substantially longer than one cycle of the free-running timer.

25. The method of claim 24, wherein determining when a transaction identifier has been stored in the scoreboard for substantially longer than one cycle of the free-running timer comprises determining if the second timer flag has been set when the free-running timer reaches the zero point.

26. The method of claim 19, wherein timing the selected duration comprises comparing a period of a free running timer to approximately the length of time since the first client sent the request.

27. The method of claim 19, wherein initiating a time-out sequence comprises notifying a fill-code generator that the transaction identifier in at least one location has remained in the scoreboard for substantially more than one period of the free-running timer.

28. The method of claim 27, wherein initiating a time-out sequence further comprises generating a fill code.

29. The method of claim 28, wherein initiating a time-out sequence further comprises transmitting the fill code to the first client.

30. The method of claim 29, wherein initiating a time-out sequence further comprises notifying the first client that a time-out has occurred.

31. A method, comprising:

storing at least one transaction identifier in at least one of a plurality of locations in a scoreboard, wherein each transaction comprises a first client requesting data from a second client in a system;

detecting approximately synchronously transaction identifiers that have been stored in the scoreboard substantially during the previous period of a free-running timer having a period approximately equal to a predetermined latency period;

determining approximately synchronously when at least one of the transaction identifiers has been stored in the scoreboard for substantially longer than one cycle of the free-running timer; and

initiating approximately synchronously a time-out sequence if the transaction identifier remains in the scoreboard substantially longer than the predetermined latency period.

32. The method of claim 31, wherein detecting transaction identifiers that have been stored in the scoreboard substantially during a previous period of the free-running timer comprises examining a first timer flag when the free-running timer reaches a zero point.

5 33. The method of claim 32, wherein detecting transaction identifiers that have been stored in the scoreboard substantially during a previous period of the free-running timer further comprises setting the first timer flag when the first timer flag has not been set and setting a second timer flag when the first timer flag has been set.

10 34. The method of claim 31, wherein determining when a transaction identifier has been stored in the scoreboard for substantially longer than one cycle of the free-running timer comprises determining if the second timer flag has been set when the free-running timer reaches the zero point.

15 35. The method of claim 31, wherein initiating a time-out sequence comprises notifying a fill-code generator that the transaction identifier in at least one location has remained in the scoreboard for substantially more than one period of the free-running timer.

20 36. The method of claim 35, wherein initiating a time-out sequence further comprises generating a fill code and transmitting the fill code to the first client.

37. The method of claim 36, wherein initiating a time-out further comprises notifying the first client that a time-out has occurred.